



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
-----------------	-------------	----------------------	---------------------	------------------

10/822,454

04/12/2004

Erez Haba

13768.1433

2824

47973

7590

08/28/2009

WORKMAN NYDEGGER/MICROSOFT

1000 EAGLE GATE TOWER

60 EAST SOUTH TEMPLE

SALT LAKE CITY, UT 84111

EXAMINER

WANG, BEN C

ART UNIT

PAPER NUMBER

2192

MAIL DATE

DELIVERY MODE

08/28/2009

PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No. 10/822,454	Applicant(s) HABA ET AL.	
	Examiner BEN C. WANG	Art Unit 2192	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 16 July 2009.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-5, 11, 12, 17, 18, 20, 22, 23, 25 and 27 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-5, 11, 12, 17, 18, 20, 22, 23, 25, and 27 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413) |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____ |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on July 16, 2009 has been entered.

2. Applicant's amendment dated July 16, 2009, responding to the Final Office Action mailed April 16, 2009 provided in the rejection of claims 1-5, 11, 12, 14, 17, 18, 20, 22, 23, and 25-27, wherein claims 1-5, 11, 12, 17, 18, 20 and 23 have been amended, claims 14 and 26 have been canceled.

Claims 1-5, 11, 12, 17, 18, 20, 22, 23, 25, and 27 remain pending in the application and which have been fully considered by the examiner.

Applicant's arguments with respect to claims currently amended have been fully considered but are moot in view of the new grounds of rejection – see *McNeely et al.* - art made of record, as applied hereto.

Claim Objections

3. Claims 1, 12, 17, and 23 are objected to because the following informalities:

- Acronyms “XML” and “XSLT” should be spelled out at the first appearance in claims.

- "... that enable the querying ...", in claim 1 at line 13, should be corrected to read -- ... that enables the querying ... --, as to overcome the typographic error.
- "... the source under.", in claim 12 at line 2, should be corrected to read -- ... the source under test. --, as to overcome the typographic error.

Appropriate correction is required (See MPEP § 608.01(b))

Claim Rejections – 35 USC § 103(a)

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

4. Claims 1, 2, 5, 12, 17, 18, 22, 23, 25, and 27 are rejected under 35 U.S.C. 103(a) as being unpatentable over McNeely et al. (Pub. No. US 2002/0162059 A1) (hereinafter 'McNeely' - art made of record) in view of Mandava (Pat. No. US 7,203,928 B2) (hereinafter 'Mandava-2')

5. **As to claim 1** (Currently Amended), McNeely discloses an application test management system that maintains fine-grained versioning of tests and their

Art Unit: 2192

relationship to software under test without sacrificing querying, filtering, and reporting, the system comprising:

a computer readable storage medium having stored thereon the following components executable by a processor:

a version component that detects versions of a source under test and versions of one or more tests that test the source under test:

- a test case file component that receives metadata that defines which versions of the one or more tests test which versions of the source under test, and stores the metadata in conjunction with test results that are generated by executing the one or more tests on the source under test, wherein metadata is also stored which indicates the version of the one or more tests and the version of the source under test to which the test results correspond (e.g., Fig. 4, elements 302 – System Under Test ; 316 – Test Plan/Case Execution Manager; 318 – Report Manager; 350 – Version Control Environment; 352 – Test Plan/Case Library; 354 – Test Results Library; [0017] - ... test plans and their associated test cases are maintained in a version-controlled environment; [0019] – The version-controlled environment may be extended to include operating software associated with each DUT (Device Under Test). As such, a test plan may specify a particular version of a test case, which may in turn specify a particular program or software version that is to be installed on a given DUT for the duration of the test. Corresponding test results may also be stored and maintained in the version-controlled

environment; [0039]; [0060] - ... test plan and test case manager 316
examines information contained in a test case or test plan related to the
operating software version required on a particular test device ...)

Further, McNeely discloses methods and systems for creating and executing sequences of interrelated test cases and providing a generalized test environment that allows complete automation of test cases (e.g., [0015]) but does not explicitly disclose other limitations stated below.

However, in an analogous art of *Method and System for Generating and Maintaining Uniform Test Results*, Mandava-2 discloses:

- storing the metadata in an XML file (e.g., Col. 4, Lines 24-30 - ... the dynamic XML file may include a test case identification, a status of the test case, and a status description);
- the test case file component further storing attributes in the XML file that enable(s) the querying of the test results (e.g., Col. 8, Lines 46-50 - ... the static XML file is configured to include entries for each and every test and test case ... the static XML file also includes a comment describing the function of each test case and test); and
- a component that uses the attributes of the XML file to transform the XML file utilizing XSLT to enable the querying of the test results based on the version of the source under test and the version of the one or more tests which correspond to the test results (e.g., Fig. 4, elements 118" – Merged Dynamic XML Results File; 120 – XSLT Interface; 124 – Report Tool; Col. 8, Lines 17-25 - ... the

uniform results are stored to storage 116 in a dynamic XML result file 118. The uniform results in the dynamic XML 118 can be viewed by a user 122 using a, Extensible Stylesheet Language (XSLT) Stylesheet interface 120; **NOTE:** McNeely teaches the version-controlled environment may be extended to include operating software associated with each DUT (Device Under Test). As such, a test plan may specify a particular version of a test case, which may in turn specify a particular program or software version that is to be installed on a given DUT for the duration of the test. Corresponding test results may also be stored and maintained in the version-controlled environment (e.g., [0019]))

Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made to combine the teachings of Mandava-2 into the McNeely's system to further provide other limitations stated above in the McNeely system.

The motivation is that it would further enhance the McNeely's system by taking, advancing and/or incorporating the Mandava-2's system which offers significant advantages for providing consistent and uniform test results easily understandable by all the developers and storing test results to storage periodically, limiting the need for re-execution to those test or test cases defined subsequent to the point of crash as once suggested by Mandava-2 (e.g., Col. 20, Lines 40-60)

6. **As to claim 2** (Currently Amended) (incorporating the rejection in claim 1), McNeely discloses the system wherein the attributes includes a pointer to the source under test (e.g., Fig. 4, elements 302 – System Under Test ; 316 – Test Plan/Case Execution Manager; 318 – Report Manager; 350 – Version Control Environment; 352 –

Test Plan/Case Library; 354 – Test Results Library; [0017] - ... test plans and their associated test cases are maintained in a version-controlled environment; [0019] – The version-controlled environment may be extended to include operating software associated with each DUT (Device Under Test). As such, a test plan may specify a particular version of a test case, which may in turn specify a particular program or software version that is to be installed on a given DUT for the duration of the test ...)

7. **As to claim 5** (Currently Amended) (incorporating the rejection in claim 1), McNeely discloses the system wherein the attributes include a pointer to a test (e.g., [0019] – The version-controlled environment may be extended to include operating software associated with each DUT (Device Under Test). As such, a test plan may specify a particular version of a test case, which may in turn specify a particular program or software version that is to be installed on a given DUT for the duration of the test. Corresponding test results may also be stored and maintained in the version-controlled environment)

8. **As to claim 12** (Currently Amended) (incorporating the rejection in claim 11), McNeely discloses the system wherein the test results are generated by a test execution component that executes the one or more tests on the source under (test) (e.g., Fig. 4, elements 302 – System Under Test ; 316 – Test Plan/Case Execution Manager; 318 – Report Manager; 350 – Version Control Environment; 352 – Test Plan/Case Library; 354 – Test Results Library; [0017] - ... test plans and their associated test cases are maintained in a version-controlled environment; [0019] – The

version-controlled environment may be extended to include operating software associated with each DUT (Device Under Test). As such, a test plan may specify a particular version of a test case, which may in turn specify a particular program or software version that is to be installed on a given DUT for the duration of the test. Corresponding test results may also be stored and maintained in the version-controlled environment; [0039]; [0060] - ... test plan and test case manager 316 examines information contained in a test case or test plan related to the operating software version required on a particular test device ...)

9. **As to claim 17** (Currently Amended), Foster discloses a test management methodology comprising:

- retrieving metadata that defines a version of source code and a version of one or more test cases that test the source code (e.g., Fig. 4, elements 302 – System Under Test ; 316 – Test Plan/Case Execution Manager; 318 – Report Manager; 350 – Version Control Environment; 352 – Test Plan/Case Library; 354 – Test Results Library; [0017] - ... test plans and their associated test cases are maintained in a version-controlled environment);
- persisting the metadata in conjunction with test results that are generated by executing the one or more tests on the source code, wherein metadata is also stored which indicates the version of the one or more tests and the version of the source code to which the test results correspond (e.g., Fig. 4, elements 302 – System Under Test ; 316 – Test Plan/Case Execution Manager; 318 – Report Manager; 350 – Version Control Environment; 352 – Test Plan/Case Library; 354

– Test Results Library; [0017] - ... test plans and their associated test cases are maintained in a version-controlled environment; [0019] – The version-controlled environment may be extended to include operating software associated with each DUT (Device Under Test). As such, a test plan may specify a particular version of a test case, which may in turn specify a particular program or software version that is to be installed on a given DUT for the duration of the test. Corresponding test results may also be stored and maintained in the version-controlled environment; [0039]; [0060] - ... test plan and test case manager 316 examines information contained in a test case or test plan related to the operating software version required on a particular test device ...);

Further, McNeely discloses methods and systems for creating and executing sequences of interrelated test cases and providing a generalized test environment that allows complete automation of test cases (e.g., [0015]) but does not explicitly disclose other limitations stated below.

However, in an analogous art of *Method and System for Generating and Maintaining Uniform Test Results*, Mandava-2 discloses:

- persisting the metadata to an XML file (e.g., Col. 4, Lines 24-30 - ... the dynamic XML file may include a test case identification, a status of the test case, and a status description);
- the test case file component further storing attributes in the XML file that enable the querying of the test results (e.g., Col. 8, Lines 46-50 - ... the static XML file is configured to include entries for each and every test and test case ... the static

XML file also includes a comment describing the function of each test case and test); and

- transforming the XML file utilizing XSLT and the attributes to enable a user to view at least one of exception patterns, trends, productivity, and success rates and management operations including at least one of selection, query, reporting, suit composition, and scheduling (e.g., Fig. 4, elements 118” – Merged Dynamic XML Results File; 120 – XSLT Interface; 124 – Report Tool; Col. 8, Lines 17-25 - ... the uniform results are stored to storage 116 in a dynamic XML result file 118. The uniform results in the dynamic XML 118 can be viewed by a user 122 using a, Extensible Stylesheet Language (XSLT) Stylesheet interface 120; **NOTE:** McNeely teaches the version-controlled environment may be extended to include operating software associated with each DUT (Device Under Test). As such, a test plan may specify a particular version of a test case, which may in turn specify a particular program or software version that is to be installed on a given DUT for the duration of the test. Corresponding test results may also be stored and maintained in the version-controlled environment (e.g., [0019]))

Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made to combine the teachings of Mandava-2 into the McNeely’s system to further provide other limitations stated above in the McNeely system.

The motivation is that it would further enhance the McNeely’s system by taking, advancing and/or incorporating the Mandava-2’s system which offers significant advantages for providing consistent and uniform test results easily understandable by

Art Unit: 2192

all the developers and storing test results to storage periodically, limiting the need for re-execution to those test or test cases defined subsequent to the point of crash as once suggested by Mandava-2 (e.g., Col. 20, Lines 40-60)

10. **As to claim 18** (Currently Amended) (incorporating the rejection in claim 17), McNeely discloses the method wherein the metadata that defines the versions of the source code and the one or more tests is retrieved from a version component that monitors changes to the source code and the one or more tests (e.g., Fig. 4, elements 302 – System Under Test ; 316 – Test Plan/Case Execution Manager; 318 – Report Manager; 350 – Version Control Environment; 352 – Test Plan/Case Library; 354 – Test Results Library; [0017] - ... test plans and their associated test cases are maintained in a version-controlled environment; [0019] – The version-controlled environment may be extended to include operating software associated with each DUT (Device Under Test). As such, a test plan may specify a particular version of a test case, which may in turn specify a particular program or software version that is to be installed on a given DUT for the duration of the test. Corresponding test results may also be stored and maintained in the version-controlled environment; [0039]; [0060] - ... test plan and test case manager 316 examines information contained in a test case or test plan related to the operating software version required on a particular test device ...)

11. **As to claim 22** (Original) (incorporating the rejection in claim 17), please refer to claim 17 above, accordingly.

12. **As to claim 23** (Currently Amended), Foster discloses a testing methodology comprising:

- loading a test case in accordance with a test case file stored in a source file;
- executing the test case on a source code under test;
- generating test results, wherein the test results are version tagged to indicate the relationships between test results, version of the test case, and version of the source code under test (e.g., Fig. 4, elements 302 – System Under Test ; 316 – Test Plan/Case Execution Manager; 318 – Report Manager; 350 – Version Control Environment; 352 – Test Plan/Case Library; 354 – Test Results Library; [0017] - ... test plans and their associated test cases are maintained in a version-controlled environment; [0019] – The version-controlled environment may be extended to include operating software associated with each DUT (Device Under Test). As such, a test plan may specify a particular version of a test case, which may in turn specify a particular program or software version that is to be installed on a given DUT for the duration of the test. Corresponding test results may also be stored and maintained in the version-controlled environment; [0039]; [0060] - ... test plan and test case manager 316 examines information contained in a test case or test plan related to the operating software version required on a particular test device ...);

Further, McNeely discloses methods and systems for creating and executing sequences of interrelated test cases and providing a generalized test environment that

Art Unit: 2192

allows complete automation of test cases (e.g., [0015]) but does not explicitly disclose other limitations stated below.

However, in an analogous art of *Method and System for Generating and Maintaining Uniform Test Results*, Mandava-2 discloses:

- saving the test results to an XML file, wherein the XML file stores metadata that defines the version of the source code and the version of the test case which were executed to generate the test results, and wherein the XML file further stores pointers to the version of the source code and the version of the test case (e.g., Col. 4, Lines 24-30 - ... the dynamic XML file may include a test case identification, a status of the test case, and a status description; Col. 8, Lines 46-50 - ... the static XML file is configured to include entries for each and every test and test case ... the static XML file also includes a comment describing the function of each test case and test); and
- employing XSLT to transform the XML file into an in memory representation of a database that enables the test results to be queried (e.g., Fig. 4, elements 118” – Merged Dynamic XML Results File; 120 – XSLT Interface; 124 – Report Tool; Col. 8, Lines 17-25 - ... the uniform results are stored to storage 116 in a dynamic XML result file 118. The uniform results in the dynamic XML 118 can be viewed by a user 122 using a, Extensible Stylesheet Language (XSLT) Stylesheet interface 120; **NOTE:** McNeely teaches the version-controlled environment may be extended to include operating software associated with each DUT (Device Under Test). As such, a test plan may specify a particular

version of a test case, which may in turn specify a particular program or software version that is to be installed on a given DUT for the duration of the test.

Corresponding test results may also be stored and maintained in the version-controlled environment (e.g., [0019]))

Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made to combine the teachings of Mandava-2 into the McNeely's system to further provide other limitations stated above in the McNeely system.

The motivation is that it would further enhance the McNeely's system by taking, advancing and/or incorporating the Mandava-2's system which offers significant advantages for providing consistent and uniform test results easily understandable by all the developers and storing test results to storage periodically, limiting the need for re-execution to those test or test cases defined subsequent to the point of crash as once suggested by Mandava-2 (e.g., Col. 20, Lines 40-60)

13. **As to claim 25** (Original) (incorporating the rejection in claim 23), Mandava-2 discloses the method further comprising publishing the test results to an enterprise data store (e.g., Fig. 3A, elements 116 and 118; Col. 8, Lines 17-26 - ... the uniform results are stored to storage 116 in a dynamic XML result file 118)

14. **As to claim 27** (Original) (incorporating the rejection in claim 23), please refer to claim **23** above, accordingly.

Art Unit: 2192

15. Claims 3, 4, 11 and 20 are rejected under 35 U.S.C. 103(a) as being unpatentable over McNeely in view of Mandava-2 and Mandava (Pat. No. US 7,210,066 B2) (hereinafter 'Mandava-1')

16. **As to claim 3** (Currently Amended) (incorporating the rejection in claim 1), McNeely and Mandava-2 do not explicitly disclose the limitations stated below.

However, in an analogous art of *Method and System for Determining Computer Software Test Coverage*, Mandava-1 discloses:

- the system wherein the attributes include a pointer to requirement for test data (e.g., Fig. 1A; Col. 2, Lines 30-46 - ... Each assertion document has a corresponding tagged assertion for each assertion in the respective specification. Each tagged assertion is defined in a markup language ...)

Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made to combine the teachings of Mandava-1 into the McNeely-Mandava-2's system to further provide the limitations stated above in the McNeely-Mandava-2 system.

The motivation is that it would further enhance the McNeely-Mandava-2's system by taking, advancing and/or incorporating the Mandava-1's system which offers significant advantages for allowing assertions in a specification document to be correlated with data in a static XML; and a user can query the assertion coverage tool of the present invention so as to determine whether a specific assertion in the specification document

has been tested, or whether a specific assertion has been tested in excess by a plurality of test cases as once suggested by Mandava-1 (e.g., Col. 28, Lines 42-61)

17. **As to claim 4** (Currently Amended) (incorporating the rejection in claim 1), Mandava-1 discloses the system wherein the attributes include a pointer to requirement (e.g., Fig. 1A; Col. 2, Lines 30-46 - ... Each assertion document has a corresponding tagged assertion for each assertion in the respective specification. Each tagged assertion is defined in a markup language ...; Col. 2, Lines) and McNeely discloses configuration under test data (e.g., [0090] - ... device configuration information ...)

18. **As to claim 11** (Currently Amended) (incorporating the rejection in claim 1), Mandava-1 discloses the system wherein the XML file is stored in a catalog with other XML files, and wherein the XML file has a hierarchical relationship with at least one of the other XML files (e.g., Figs. 3E-1; 3E-2; 3F-1; Col. 26, Line 61 through Col. 27, Line 22 - ... a test suite structure ...)

19. **As to claim 20** (Currently Amended) (incorporating the rejection in claim 17), McNeely discloses the method wherein the attributes comprises a pointer to at least one of the source code (e.g., [0019] – The version-controlled environment may be extended to include operating software associated with each DUT (Device Under Test). As such, a test plan may specify a particular version of a test case, which may in turn specify a particular program or software version that is to be installed on a given DUT for the

Art Unit: 2192

duration of the test), further Mandava-1 discloses a requirement under test (e.g., Fig. 1A; Col. 2, Lines 30-46 - ... Each assertion document has a corresponding tagged assertion for each assertion in the respective specification. Each tagged assertion is defined in a markup language ...), and furthermore McNeely discloses a configuration under test (e.g., [0090] - ... device configuration information ...)

Conclusion

20. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Ben C. Wang whose telephone number is 571-270-1240. The examiner can normally be reached on Monday - Friday, 8:00 a.m. - 5:00 p.m., EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on 571-272-3695. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Art Unit: 2192

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Ben C Wang/

Ben C. Wang

Examiner, Art Unit 2192

/Michael J. Yigdall/

Primary Examiner, Art Unit 2192